

II in a Mac

Apple II Emulation for Macintosh

COMPUTER applications, Inc.

Copyright

This manual and the software described in it are copyrighted with all rights reserved. Under the copyright laws, this manual of the software may not be copied, in whole or part, without written consent of COMPUTER:applications, except in the normal use of the software or to make a backup copy. The same proprietary and copyright notices must be affixed to any permitted copies as were affixed to the original. This exception does not allow copies to be made for others, whether or not sold, but all of the material purchased (with all backup copies) may be sold, given, or loaned to another person. Under the law, copying includes translating into another language or format.

You may use the software on any computer owned by you, but extra copies may not be made for this purpose. A multi-use license may be purchased to allow the software to be used on more than one computer owned by the purchaser. Contact COMPUTER:applications for information.

Limited Warranty on Media and Manuals

If you discover physical defects in the media on which this software is distributed, or in the manuals distributed with the software, COMPUTER:applications will replace the media or manuals at no charge to you, provided you return the item to be replaced with proof of purchase to COMPUTER:applications during the 90 day period after you purchased the software.

ALL IMPLIED WARRANTIES ON THE MEDIA AND MANUAL, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN DURATION TO NINETY (90) DAYS FROM THE DATE OF THE ORIGINAL RETAIL PURCHASE OF THIS PRODUCT.

COMPUTER:applications MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS SOFTWARE, ITS QUALITY, PERFORMANCE, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS SOFTWARE IS SOLD "AS IS", AND YOU, THE PURCHASER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND PERFORMANCE.

IN NO EVENT WILL COMPUTER:applications BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT IN THE SOFTWARE OR ITS DOCUMENTATION, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No COMPUTER:applications dealer or agent is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights,

]] In a Mac

© 1985 by
COMPUTER:applications, Inc.
12813 Lindley Drive
Raleigh, NC 27614
(919) 846-1411

written by
Randy Ubillos

About This Manual

This manual describes the operation of the][in a Mac™ Apple II simulator. Chapter 1, "Learning][in a Mac", gets you started running Apple][programs and learning how to operate][in a Mac.

Chapter 2 describes each of the menu options in detail. Chapter 3 shows the procedure for transferring software into][in a Mac. Chapter 4 details the features of each of the Apple II slots which is emulated by][in a Mac. Chapter 5 provides technical information for those users who wish to delve deeper into the workings of][in a Mac and the Apple II.

Appendix A lists those manuals which are recommended to those who wish to learn more about the Apple II. A list of trademarks also appears there. Appendixes B through E provide brief descriptions of the commands used with the major pieces of software for the Apple, applesoft, Integer BASIC, DOS 3.3 and ProDOS.

Packing List

Inside your][in a Mac package you should receive:

- 1][in a Mac master disk
- 1 Masters disk with DOS 3.3 and ProDOS
- 1 5 1/4" Communications disk
- 1 User's manual
- 1 Registration card
- 1 Information card on the][in a Mac library disks

Table of Contents

1	Chapter 1: Learning][in a Mac
1	Starting Up
2	Sample Applesoft program
4	Working with Disks
5	Making a Backup
6	Chapter 2: What's On the Menus?
6	Apple Menu
6	File Menu
8	Edit Menu
9	Disks Menu
11	Option Menu
13	Help Menu
14	Chapter 3: Transferring Software
14	Cable Transfer
15	Phone Transfer
17	Chapter 4: Hardware Emulation
17	Slot 1: Printer Interface
18	Slot 2: Communications Card
19	Slot 3: 80 Column Card
21	Slot 4: Clock Card
22	Slots 5 & 6: Floppy Disk Drives
23	Slot 7: RAM Disks
25	Chapter 5: Programmer's Reference
25	I/O Soft-Switches
26	Status Display
29	Compatibility
30	Calling 68000 Subroutines
31	Illegal Opcodes
32	Appendix A: References
32	Recommended Reading
32	Trademarks
33	Appendix B: Applesoft Reference
38	Appendix C: Integer Basic Reference
42	Appendix D: DOS 3.3 Reference
44	Appendix E: ProDOS Reference

Learning II in a Mac

To begin using II in a Mac™, insert the II in a Mac Master disk and turn on the Macintosh. In a few moments the desktop will appear. You will see the II in a Mac icon, along with the icon for the Apple disk called 'Boot Disk'. Double click on the II in a Mac icon to start the program.

After a short delay, the II in a Mac desktop will appear and display the tools available for your use: the Apple screen, the keyboard, a joystick, 4 disk drives, a status panel, a clock and a printer.

The Apple II screen will first display 'APPLE II', and then the II in a Mac Main Menu will appear. The first option allows files to be moved to and from the Macintosh environment. The next two options allow the user to enter either of II in a Mac's built in languages.

At this point, a little history is in order. When the Apple was first released, it had one of the first BASIC interpreters for a microcomputer. It was called Integer BASIC, and as the name suggests, could only handle integers. Many programs were written with this BASIC, since it was the only thing available at the time.

As the Apple became more popular, a new floating point BASIC was released. It was called Applesoft. This BASIC had a much wider variety of commands, and supported the Apple's high resolution (hires) graphics. At first this BASIC was available on a plug in card, but as its popularity grew, it became the built-in BASIC for the Apple.

Today most programs are written in applesoft, but a large number of integer BASIC programs still exist. To make sure that all programs can be run, II in a Mac provides the user with both versions of BASIC. When learning a first BASIC, it is recommended that the user stay with Applesoft BASIC, since it is just as easy to use as Integer, while being much more powerful.

The best way to learn][in a Mac is with some examples. We will show a short programming example in applesoft, and then demonstrate how to run existing programs.

Remember, the first option on the Help menu is a full on-line reference section which can answer many questions. It is always just a click away, so please use it when in doubt.

For this example, we want to use Applesoft BASIC, so select option 2 from the][in a Mac Main Menu, which should be on the screen. The screen will clear and a message will be displayed indicating that we have exited to the applesoft language. The bracket character (]) will be displayed, with a white cursor next to it. This bracket is the symbol for applesoft BASIC (a greater than sign is the symbol for Integer BASIC). At this point, applesoft is awaiting a command. Type in the following:

```
PRINT 6+5*3    <RETURN>
```

the RETURN key must be pressed at the end of each line. This tells applesoft when a line is complete. In the following examples, it is assumed that the RETURN key is to be pressed at the end of each line.

In this example the number 21 is printed on the screen below the line we typed, and the bracket is once again displayed. The line we typed simply asked applesoft to print out the result of a calculation. If a number is added at the beginning of a line, the line is stored and can be executed later as a program. Lets try that now by typing in the following lines (don't forget to press RETURN after each line).

```
10 FOR A=1 TO 10  
20 PRINT A  
30 NEXT A  
40 END
```

This simple program will display the numbers from one to ten on the screen. To make sure that the program has been typed in properly we can list the program on the screen. This

is done with the command:

LIST <RETURN>

This displays the entire program on the screen. To see the program work, type in the command:

RUN <RETURN>

If the program has been typed in correctly the numbers from 1 to 10 will be displayed on the screen.

This is a simple program, but it could have been much longer. A complete tutorial on applesoft would take up too much space here, but a brief reference on the commands of applesoft is listed in Appendix B. Apple Computer publishes a book called the *Applesoft Tutorial* which is an excellent reference for users who wish to write programs in applesoft.

For many people, running existing programs will be the main concern. We can try this now. The first order of business is to display the files on the disk. This is done with the command:

CATALOG <RETURN>

a list of all of the files on the current disk will be displayed on the screen (If the listing is longer than one page, the listing will pause after each page, waiting for the space bar to be pressed). The letter displayed in the second column denotes the type of file. Those which have an 'A' next to them are applesoft programs. One of these is "BRIAN'S THEME". To run this program we simply type:

RUN BRIAN'S THEME <RETURN>

After pressing RETURN when prompted by the program, this program displays a variety of patterns on the screen. Pressing any key exits this program.

You may experiment with the other programs on this disk in a similar fashion.

Working With Disks

As stated previously,][in a Mac stores the information from Apple II diskettes in files on Macintosh diskettes. These files display the icon of a diskette from the Finder. These files may be moved around, or have their names changed as desired. On the Apple II, a disk could be 'booted' by inserting it and turning the power on. With][in a Mac the same result can be obtained by double clicking the desired disk file. When this is done,][in a Mac will run and the disk which was selected will be 'inserted' into a drive and booted.

When the][in a Mac icon is double clicked, it looks for a file called 'Boot Disk' on the same disk which it is on. If this file is located, it is 'booted' as described above. On the][in a Mac master, this disk contains some utility programs. This disk can actually be any][in a Mac disk file, named as 'Boot Disk'. This allows a user to create his own turnkey system.

The 'Masters Disk' supplied with][in a Mac contains a DOS 3.3 master disk, as well as a ProDOS master disk (the two Apple II operating systems). These disks may be booted by inserting them into a drive and double clicking on them.

If][in a Mac is already running, a new disk can be booted in another fashion. Use the options on the Disks menu (described in chapter 2) to remove the current disk from slot 6, drive 1. Then insert the disk which is to be booted into that drive. Then type the command:

PR#6 <RETURN>

This will boot the new disk. (Disks can also be booted from slot 5 in this manner by typing PR#5 with a disk inserted in slot 5, drive 1). PR#6 (or 5) is the standard Apple II command for booting a disk.

Making a Backup Copy

It is a good idea to make a backup copy of your][in a Mac disk. Even though][in a Mac is protected, you can make a backup copy. The original disk will only be required once while][in a Mac is starting up for verification. This is also true for those users who have hard disks.][in a Mac can be moved to a hard disk and the original disk will only be required once while][in a Mac is starting up.

To make a backup, first boot the][in a Mac master disk. Then insert the disk which is to be used as a backup. If the disk is a new disk, perform the initialization in the normal fashion. If the disk contains files, use the 'Clear' option to remove them. Now move the mouse to the left of the][in a Mac icon. Then press and hold the mouse button while dragging it to the right side of the 'Boot Disk' icon. Then release the mouse and the three icons will appear highlighted. Then press the mouse on one of these three icons and drag it to the new disk. All three icons should follow along. Release the mouse and the Macintosh will proceed to copy all of the files.

Now the new disk can be used instead of the original. Be sure to store the original in a safe place.

Those user's with a hard disk can simply transfer the files '][in a Mac', 'Help File', 'Screen', and 'FP.BIN' to their hard disk to use][in a Mac from there.

NOTE: Registered owners can purchase a backup disk for \$5.00. Send a check or money order along with your registration card to receive your backup disk. Any disk which is damaged will be replaced for \$5.00 for as long as you own the disk. (Physical defects during the first 90 days will be replaced free of charge).

What's On the Menus?



About][in a Mac

The first item on this menu displays the][in a Mac logo screen. Clicking the mouse button causes the display to disappear.

][in a Mac is shipped with some standard desk accessories, like the calculator and the alarm clock. These accessories will show up here, and can be used in the normal Macintosh fashion. If other accessories are added to the disk, they will also show up here.

File



Reset Apple][⌘.

This option is functionally identical to the Reset key on an Apple][. The keyboard equivalent is command-period. This option has the effect of stopping the program currently running and returning control to the keyboard.



Re-Start][in a Mac

It is possible for a program to take complete control of][in a Mac. It is also possible for a program to modify the internal code of the simulated Apple][to the point where normal operation no longer takes place. This option re-loads all of the internal code and re-boots the disk in slot 6, drive 1.



Modem Settings

This option allows the configuration of the modem port to be selected through a series of buttons. You may choose any of the displayed options for the port. Clicking the "Ok" button causes the selected options to become active. The "Cancel" button exits this display without changing the settings.



Inside][in a Mac

The internal slot arrangement of][in a Mac is displayed with this option. Slots 1 through 6 have fixed assignments. The two RAM disks in slot 7 are available if memory permits. This display shows the name of the RAM disk dimmed if it is not active.



Quit to Finder

This exits][in a Mac and returns to the Macintosh Finder. Be aware that any data in the RAM disks is lost when this option is selected.

Edit

Undo

Cut

Copy

Paste

Clear

the Notepad desk accessory with the Paste option allows an easy keyboard macro system.



Select Text



Select Graphics

These are the standard Macintosh editing options. The first three are only active when a desk accessory is open. The Paste option will take any text on the clipboard and type it in as if it were typed from the keyboard. The Clear option removes any information which is on the clipboard. Using

These options allow text or graphics to be copied from ⌘ in a Mac screens onto the Macintosh clipboard.

The first option enters the text selection mode. In this mode the cursor will

change to the I-beam cursor whenever it is within the Apple ⌘ screen. To select a section of displayed text, move to the beginning of the area desired. Then press and hold the mouse button while dragging the cursor to the other end of the desired text. As the mouse is moved, the selected area will be highlighted. To select a different area, simply release the mouse and start again. When the desired area is selected, choose Copy from the Edit menu. The Select Text option may be chosen again to cancel this mode.

Selecting graphics is handled in a similar fashion. When selected, the cursor may be dragged over the area to copy to the clipboard. A rectangle is displayed to show the selected area. Use the Copy option to copy the area to the clipboard. Selecting the Graphics option again cancels this mode.

Disks



Insert a disk into Slot 6, Drive 1...



Insert a disk into Slot 6, Drive 2...



Insert a disk into Slot 5, Drive 1...



Insert a disk into Slot 5, Drive 2...

These four options control the][in a Mac disk drives. Apple][disketes are stored on the Macintosh as files. These files are seen from the finder as small diskettes. In order to use a diskette, it must be assigned to one of the four 'disk drives' which][in a Mac emulates. This assignment process is called 'insertion', because it corresponds to the act of inserting a 5 1/4" diskette into an Apple][.

When one of these options is selected, the standard Macintosh display prompting for the choice of a file is presented. Disks may be ejected and swapped to select the desired file. When the Open button is clicked the image of the disk drive on the screen will show with its door shut, with the name of the file assigned to it displayed below the door. The disk is then available for use by Apple][programs using the standard file commands.

The options on this menu will change as disks are inserted. When a drive has a disk in it, the menu item is used to remove that disk from the drive. This displays an open door on the screen and removes the name below it.

These options can be used at any time to manage the disk drives.



Create an Apple II diskette...

This option is used to make a new disk file. Since the contents of each Apple II diskette used by II in a Mac is contained in a Macintosh file, it is necessary to create these files before any Apple II data can be stored. This option displays the standard Macintosh file selection box and asks for the name of the new disk and the drive where it should be created. While the disk is being created, a message is displayed. This process can take up to one minute.

The disk may be created on a floppy drive or any other drive connected to your Macintosh, such as a hard disk.



Receive an Apple II diskette...

To move an Apple II diskette to the Macintosh, this option is used. Its operation is covered in detail in chapter 3.

Options



Use Joystick

][in a Mac provides the equivalent of a joystick controlled by the mouse. Selecting this option causes the cursor to disappear, and the joystick on the screen becomes active, moving with the mouse.

The Apple][program will react as if this were a real joystick. The buttons on the joystick are controlled by the mouse button. The mouse button acts as button 0, with the shift key pressed, it acts as button 1. The two buttons on the screen image of the joystick will light up according to the position of the shift key.

To exit from this mode, hold down the command key and click the mouse. A reminder of this will appear at the top of the screen while in joystick mode.



Reverse horizontal



Reverse vertical



Swap Axis

The next three options control how the joystick responds to the user. Since there are a variety of joysticks available for the Apple][, each with its own characteristics, these options allow the user to configure the joystick for particular actions. These options toggle their respective controls each time that they are activated, and may be used at any time.



Use Keyboard

[[in a Mac allows the keyboard on the screen to be used for text entry. After selecting this option, the cursor is confined to the area around the keyboard. Clicking the mouse over the keys causes the corresponding character to be typed. The shift and control stay highlighted until either they are pressed again, or until another key is pressed.

To exit this mode, hold down the command key and click the mouse. A reminder of this is displayed in the upper right hand corner of the screen while in keyboard mode.



Disable Keyboard Buffer

When keys are typed faster than a program can process them, [[in a Mac can remember them until they are needed. For most programs this is fine. There are, however, some games which will not operate properly with this option selected. The symptom of this problem is no keyboard response. In these cases, simply turn off the keyboard buffer until finished using the particular program.



Turn sound off

This option allows the sound generated by [[in a Mac to be turned on and off.

Programs which perform a large amount of sound generation will run faster if the sound is turned off.

Help



User Manual

]] in a Mac provides an on-line reference section which describes many of its features. When selected this options shows a table of contents for the reference section. An item may be picked directly from the list and the page for that subject will be displayed. The 'Next' and 'Previous' buttons at the bottom of the screen may be used to move forward and backward through the pages. The 'dog-ears' on the pages may also be used to flip from one page to another. The 'Index' button displays the table of contents at a single press.

To exit from the help screen, press the 'Cancel' button, or click on the close box in the upper left corner.



Product Support

This option displays the address and phone number for COMPUTER:applications, Inc. This is where all questions and/or comments should be addressed.

Also displayed is the backup policy for registered owners, and a list of the library disks which are available for use with]] in a Mac.

Chapter 3: Transferring Software

Software for][in a Mac can either be entered from the keyboard or transferred from another source. When transferring from another source, the process can be done with a direct line to an Apple II, or over the phone from any information source. These two methods are described separately in this chapter.

Cable Transfer

When an Apple II is available, the cable transfer method provides a quick and easy solution to software acquisition. This system requires an Apple II or //e with either an Apple Super Serial Card or a California Computer Systems serial card (Some other cards may be compatible, if they emulate either of the above cards and support 19.2k baud). An Apple //c can also be used with the proper cable. With some cards the imagewriter cable can be used, for others a cable must be made (the proper pinouts for this cable are shown on the next page).

The actual procedure for transferring a disk is very simple. First connect the Macintosh serial port to the Apple II. Then insert the][in a Mac Communications disk in the Apple II and turn on the Apple. When the transfer screen is shown, remove the Communications disk from the drive.

Now select 'Receive an Apple][diskette...' from the Disks menu in][in a Mac. This will prompt for the name of the file to be received, and the disk which it should be placed on. After typing in the name, press RETURN. A box will appear showing that the new disk is being created. When this is complete, start the Apple II transfer by simply pressing the space bar on the Apple II. Then click the Ok box on the Macintosh. The transfer will begin, with a visual status of the time remaining showing on the Macintosh.

When the transfer is complete, a noise will be generated and the Macintosh will wait for the 'Ok' button to be clicked for confirmation. If it is desired to transfer more disks, this process can be repeated as many times as desired.

Note: The Super Serial Card is automatically set to 19.2k baud, on any other card, it is up to the user to make sure that the baud rate is set correctly before starting the transfer.

Warning: Never insert or remove cards from an Apple II with the power on! Permanent damage may result to both the card and the computer. Always be sure the power is OFF before removing any cards!

Cable Connections

The pinouts for the Macintosh serial connector are shown below. If you have any doubts about making a cable, contact your local Apple dealer.

Pin 1 - Ground

Pin 5 - Transmit Data (output)

Pin 9 - Receive Data (input)

Connect pin 8 to pin 3 on the Macintosh end of the cable.

Phone Transfer

There are many information systems accessible through the phone lines which contain Apple software.][in a Mac provides the facility to access this tremendous base of software. The procedure for receiveing software by phone is divided into two parts.

The first step is to receive the data as a Macintosh file. This is done with the FreeTerm program. Freeterm is a public domain program provided free of charge for those users who do not have any communications software. Those users who have communications software can use their own, or they may prefer to use FreeTerm, since it has many advanced features.

After downloading the desired software to a Macintosh disk, it must be moved into the][in a Mac environment. This is done by selecting option 1 from the][in a Mac Main Menu which is displayed when the 'Boot Disk' on the][in a Mac Master disk is booted.

This option first asks in which direction the transfer is to take place. Files can be moved into the][in a Mac environment, or they can be moved out to the Macintosh environment. This allows any information to be freely shared between the two environments.

After choosing a direction, the program asks for the name of the file. If the file is being moved from the Macintosh environment, the program asks whether the resulting Apple II file should be considered a binary file or a text file. The conversion process then begins. After it is finished, the option menu is re-displayed, allowing any number of files to be converted back and forth.

When finished converting files, select option 3 to return to the][in a Mac Main Menu.

Note: For those users desiring technical details about the process used to move files to and from the Macintosh environment, Chapter 5 describes the "Pinhole Interface" which allows these transfers to take place.

FreeTerm is a public domain program written by William Bond. It is provided free of charge for use with your modem. The program may be freely reproduced as long as the credit notice in the "About FreeTerm..." menu remains intact.

Chapter 4: Hardware Emulation

In order to provide complete emulation of the Apple II,][in a Mac simulates the operation of the internal slots found inside the Apple II. Each slot has a unique purpose, as displayed with the 'Inside][in a Mac' menu option in the File menu. Each of the slots is described in detail here.

Slot 1 - Printer Interface

On an Apple II, a printer is connected through a card plugged into slot 1.][in a Mac simulates this card using the imagewriter as the printer. The standard Apple command:

PR#1 <RETURN>

activates the printer, causing everything which is displayed on the screen to also be displayed on the printer. This means that programs can be listed for easy editing, or mailing lists can be printed from Apple II software. Any normal printer functions are supported by][in a Mac.

To turn the printer off, type in the command:

PR#0 <RETURN>

While text is printed to the imagewriter, a miniature image of the printed material is shown on the paper exiting from the top of the picture of a printer in the upper right section of the][in a Mac screen.

The PR#1 and PR#0 commands shown above can only be used while either the ']' or '>' prompts are displayed. While a program is running, it is up to the program to activate or de-activate the printer as necessary.


Pressing RESET (command-period) will also turn off the printer.

Slot 2: Communications Card

On an Apple II, any serial communications were normally performed through a card inserted in slot 2.][in a Mac emulates this card, using the Macintosh serial port as the external connector.

Any software which uses this slot for communications, will talk to whatever device happens to be connected to the serial port on your Macintosh, for example, a modem.

The settings for the serial port may be adjusted using the 'Modem Settings' option on the File menu. The display below will appear and allow you to make the proper selection with the mouse. When the settings are correct, click the 'OK' button. The 'Cancel' button can be used to leave this screen without making any changes.



Select Modem Setting

Baud Rate	Data Length	
<input type="radio"/> 19200	<input checked="" type="radio"/> 8 bits	<input type="radio"/> 7 bits
<input type="radio"/> 9600		
<input type="radio"/> 4800	Parity	Stop Bits
<input type="radio"/> 2400	<input checked="" type="radio"/> None	<input checked="" type="radio"/> 1
<input checked="" type="radio"/> 1200	<input type="radio"/> Even	<input type="radio"/> 1.5
<input type="radio"/> 600	<input type="radio"/> Odd	<input type="radio"/> 2
<input type="radio"/> 300		
<input type="button" value="Ok"/>		<input type="button" value="Cancel"/>

Slot 3: 80 Column Card

The original Apple II could only display 40 columns of text across the screen. This is sufficient for most programs, especially graphics oriented ones. However, many business applications need to put more information on the screen at a time. This need caused the development of several types of 80 column cards for the Apple II. They plugged into slot 3 and connected to an external monitor, displaying 80 columns of text on the screen.

To allow these programs to run,][in a Mac provides the equivalent of an 80 column card installed in slot 3. It can be activated with the command:

`PR#3 <RETURN>`

After this command is issued, a large window will open over the][in a Mac desktop. All normal screen operations will work on this new screen, except that it is 80 columns wide instead of 40, and supports text only.

When the 80 column card is turned on, programs will list in a wider format, making program entry easier.

To turn the 80 column card off, click the close box in the upper left hand corner. This removes the 80 column window from the screen and allows the desktop below to become visible again.

The 80 column display can also be de-activated by pressing RESET (command-period).

Some software may ask the type of 80 column card being used, since there are several popular brands on the Apple, each acting slightly differently. The][in a Mac 80 column card acts most like a Videx™ Videoterm™ card. Many programs, however, do not need to ask this question, and should operate normally.

Programming with the 80 Column Card

The following special sequences can be printed from a basic program to control the operation of the 80 column card.

<u>Sequence</u>	<u>Action</u>
chr\$(8)	Moves the cursor one space to the left.
chr\$(10)	Moves the cursor down one row. If the cursor is on the last row, the screen is scrolled up one line.
chr\$(11)	Clears the display from the cursor position to the end of the screen.
chr\$(12)	Clears the screen and places the cursor in the upper left hand corner.
chr\$(13)	The carriage return, moves the cursor to the beginning of the next line on the screen.
chr\$(21)	Turns off the 80 column display.
chr\$(25)	Moves the cursor position to the upper left hand corner of the screen.
chr\$(28)	Moves the cursor forward one space.
chr\$(29)	Clears all characters from the cursor position to the end of the current line to blanks.
chr\$(30) + chr\$(x+20) + chr\$(y+20)	Moves the cursor to the position specified by the value of x and y.
chr\$(31)	Moves the cursor up one line. If the cursor is at the top of the screen, it does not move.

Also the commands INVERSE and NORMAL can be used to control the appearance of the text displayed on the screen. INVERSE causes text to be displayed as white characters on a black background, while NORMAL causes them to be displayed as black characters on a white background, which is the default.

Slot 4 - Clock Card

[[in a Mac provides software which makes the Macintosh internal clock appear as an Apple clock installed in slot 4. This clock is used automatically by ProDOS to time and date stamp files, and it is used by several commercial programs. Normally programs which use the clock locate it automatically, so no user intervention is necessary.

The current time is always displayed on the clock shown on the [[in a Mac desktop. To set the time, use the control panel desk accessory.

Programming with the Clock

The [[in a Mac clock can receive as well as generate characters. To use the clock, a character is sent to it (using PR#4). This tells the clock what format to use to return the time. Then use IN#4 and an input statement to read the time value into a string for use with a program. The following chart describes the format generated for each character which can be used with the clock:

" "	MO/DD HH:MI:SS.WYY
	12/14 15:30:23.383
" :	W MO/DD/YY HH:MI:SS
	3 12/14/83 15:30:23
" %"	WWW MMM DD HH:MI:SS PM
	WED DEC 14 03:30:23
" & "	WWW MMM DD HH:MI:SS
	WED DEC 14 15:30:23
" #"	MO,OW,DD,HH,MI,SS
	12,03,14,15,30,23
" > "	WWW MMM DD HH:MI:SS PM
	WED DEC 14 03:30:23 PM
" < "	WWW MMM DD HH:MI:SS
	WED DEC 14 15:30:23

MO	Month	DD	Day	YY	Year
HO	Hour	MI	Minute	SS	Second
WWW	Weekday	MMM	Month	OW	Day

Slots 5 & 6 - Disk Drives

The Apple computer uses 5 1/4" diskettes for information storage.][in a Mac stores its information in files which exist on Macintosh 3 1/2" disks. To use a disk on an Apple, a diskette is inserted a drive, and the door is closed. With][in a Mac, the proecedure is basically the same. The Disks menu has four options which allow the 'disk' files from a Macintosh disk to be 'inserted' into one of the four drives which are displayed on the][in a Mac screen. This process assigns a particular 'disk' file to a drive. Any accesses to that drive cause data to be read or written from the file assigned to it.

The disk 'files' which are assigned to the four drives may be on up to four different Macintosh disks, or in different drives. This includes any hard disk which may be connected to the Macintosh. The normal disk swapping messages will be displayed as necessary if a program accesses a drive whose assigned file is not currently in a drive.

Disks may be inserted or removed from the drives at any time. When a disk is in a drive, the door will appear closed, and the name of the file which is assigned to it will be displayed below the door. When the disk is removed, the door will open and the name will be removed.

The 'lights' on the disk drives on the screen will light up as a given drive is accessed.

Slot 7 - RAM disks

A RAM disk is a section of internal memory which is set aside for information storage in a format similar to that used on a disk drive. The Apple II did not have any built in RAM disks, but several manufacturers of RAM cards for the Apple provided software with their products which provided this feature. When Apple produced ProDOS they added software which made the extra memory in an Apple //e or //c appear as a RAM disk.

DOS 3.3

[[in a Mac provides two different RAM disks. The first is for DOS 3.3, and acts like a disk drive in slot 7. This means that a command like

```
CATALOG,S7 <RETURN>
```

will show a directory of the RAM disk. This RAM disk can hold 48k of data for any purpose. All of the normal filing commands will work on this drive just as they would on a normal diskette. The FID program on the DOS 3.3 Master disk is used to display the amount of free space on this or any other drive. To move data onto this disk, or from it to another disk, also use the FID program.

The information on this drive is retained until [[in a Mac is turned off. This means that you can go from DOS to ProDOS and back again without losing the data on this disk.

ProDOS

The second, completely separate, RAM disk is for ProDOS. This drive is also connected to slot 7. This means that the command:

```
CATALOG,S7 <RETURN>
```

displays the directory of the disk. This disk also responds to the standard ProDOS RAM disk name of /RAM. Up to 48k of data can be stored on this drive.

As with the DOS RAM disk, the data on this drive is retained even if a different operating system is used between accesses to the drive.

It is important to note that the information stored on a RAM disk is lost when you exit from][in a Mac. This means that any desirable information must be moved to a 'real' disk before exiting.

Each time that][in a Mac starts up the RAM disks are erased and made ready for information storage. If there are any other programs in memory using up space, there may not be enough room for the][in a Mac RAM disks.][in a Mac tries to attach one or both of the disks when possible. To see if there was enough memory, the 'Inside][in a Mac' option on the File menu is used. The information for slot 7 is:

	DOS RAM-Disk	
Slot 7 -	ProDOS RAM-Disk	

In the display above, both RAM disks are active. If either one is not, the corresponding item on the display will appear dimmed. In this case any operations performed on the drive will result in an error.

If one of the RAM disks is disabled and it is desired to use it, the program which is using up the extra memory must be removed. This will free up additional space and allow the drive to be attached and used.

Chapter 5: Programmer's Reference

This chapter contains technical information for those user's who will be programming in assembly language with `[[` in a Mac. Most users will wish to skip this section and proceed to the appendixes.

The *Apple `[[` Reference Manual* from Apple computer provides a valuable reference guide to the inner hardware of the Apple `[[`.

Soft-Switches

The Apple II used a special system for talking to internal hardware functions called memory mapped I/O. This means that all of the hardware functions are performed by accessing particular memory addresses. Most of these hardware addresses are emulated by `[[` in a Mac. Those that are not are those which do not apply, such as the Apple cassette port which have no equivalent on the Macintosh.

When writing machine language programs, there are no special considerations to be made except for timing. When writing programs for an Apple it is possible to use the 6502 data sheets to determine the amount of time each instruction will take to execute. This is normally used in timing loops. With `[[` in a Mac, the timing of the instructions is different since each one is actually being emulated by the Macintosh's 68000 processor. In some cases it may be possible to use the built in clock if a resolution of 1 second is sufficient.

The decimal mode of the 6502 is supported by `[[` in a Mac, but programs using it may run at a more reduced rate. This is because the SED and CLD (set decimal mode and clear decimal mode) instructions take significantly longer to execute than most other instructions. The fourth indicator light on the `[[` in a Mac status display corresponds to the current decimal flag setting.

Status Display

The status display on the][in a Mac screen shows the state of several internal functions. They are as follows:

Language Card Indicators Decimal Mode



Annunciators 0-3

Hardware Addresses

The following is a list of all of the hardware addresses which have an effect for][on a Mac. Some locations are controlled by access only (read or write) while others have different functions depending on whether the location is read from or written to.

<u>Hex Address</u>	<u>Effect</u>
--------------------	---------------

Internal Functions

C000-C00F	Read - loads the current keyboard value.
C010-C01F	Clears the current keyboard character.
C030-C03F	Clicks the speaker.
C050	Sets graphics mode.
C051	Sets text mode.
C052	Sets full screen graphics.
C053	Sets mixed text/graphics mode.
C054	Sets hires page 1.
C055	Sets hires page 2.
C056	Sets lores graphics mode.
C057	Sets hires graphics mode.
C058	Clears annunciator 0.
C059	Sets annunciator 0.
C05A	Clears annunciator 1.
C05B	Sets annunciator 1.

C05C	Clears annunciator 2.
C05D	Sets annunciator 2.
C05E	Clears annunciator 3.
C05F	Sets annunciator 3.
C061	Read- loads the state of button 0 in bit 7.
C062	Read- loads the state of button 1 in bit 7.
C064	Read- loads the value of PDL(0)
C065	Read- loads the value of PDL(1)

Slot 0 - Language Card

C080	Enables language card, bank 2.
C081	Disables language card, two accesses allows reading from ROM, writing to RAM, bank 2.
C082	Disables language card.
C083	Two accesses enables language card read/write, bank 2.
C088	Enables language card, bank 1.
C089	Disables language card, two accesses allows reading from ROM, writing to RAM, bank 1.
C08A	Disables language card.
C08B	Two accesses enables language card read/write, bank 1.

Slot 1 - Printer interface

C090	Write- sends a character to the imagewriter.
------	--

Slot 2 - Communications interface

C0A0	Read- load status of port: bit 0 - receive data ready bit 1- ready to transmit character
C0A1	Write- sends a character to the serial port.
C0A1	Read- loads the next character from the read buffer. Up to 512 characters can be buffered.

(All even addresses from C0A0 to C0AF are the same as C0A0, all odd addresses are like C0A1).

Slot 3 - 80 Column Card

C0B0	Selects bank 0 of the 80 column display to be mapped into \$CC00-\$CDFF.
C0B1	Write- sends a character to the 80 column display, activating it if necessary. This location uses the inverse flag to specify how the character will appear.
C0B2	Write- sends a character to the 80 column display. Bit 7 determines whether the character will be inverse or normal.
C0B4	Map bank 1 of the card into CC00-CDFF.
C0B8	Map bank 2 of the card into CC00-CDFF.
C0BC	Map bank 3 of the card into CC00-CDFF.
C0BD	Read- Bit 7 is set if the close box is pressed.
C0BE	Turns the cursor on.
C0BF	Turns the cursor off.

Slot 4 - Clock Card

C0C1	Read- Year (0-99)
C0C3	Read- Month (1-12)
C0C5	Read- Day (1-31)
C0C6	Read- Hour (1-12)
C0C7	Read- Hour (0-23)
C0C9	Read- Minute (0-59)
C0CB	Read- Second (0-59)
C0CD	Read- Day of week (0-6)
C0CF	Updates the time/date locations above.

Slot 7 - "Pinhole" File System

C0F0-C0F1	Address of file name.
C0F2	Open Mac file.
C0F3	Close Mac file.
C0F4-C0F7	Type of file for create, or position desired.
C0F8	Position read/write pointer.
C0F9	Create Mac file.
C0FC-C0FD	Error code for last operation.
C0FE	Write- send character to Mac file.
C0FF	Read- get next character from Mac file.

Compatibility

While][in a Mac is almost functionally identical to an Apple II, there are a few differences. The most obvious is that][in a Mac runs at approximately 35-40% of the speed of an Apple II. This means that software will run slower than on an Apple II. For many applications however, this will not be a big factor, since many programs spend much of their time waiting anyway.

Programs which read the joystick directly, instead of using the monitor routines, will have varying results. Usually the program will respond as though the joystick is at full swing, either left or right, depending on which side of center the][in a Mac joystick is on. Many times these same games are written so as to smooth out drastic movements of the joystick. This can compensate for the positioning inaccuracy and allow the game to be played anyway. To assure proper joystick operation, always use the monitor routine at \$FB1E to read the joystick.

Those programs which use sound will not sound the same. To speed up the operation of programs which use a great deal of sound, the 'Turn Sound Off' option on the 'Options' menu can be used. When writing programs for][in a Mac, adjust the timing and length of sound routines according to the observed sound versus the desired sound.

][in a Mac adds the ability to cut and paste to the Macintosh clipboard. This process will work with any program. Cutting graphics is normally used with a program like MacPaint for displaying a graphic from][in a Mac within a document. The text copy system allows any section of the screen, either 40 or 80 column, to be copied to the clipboard for use by any other Macintosh application. The text can even be pasted back into][in a Mac. This feature is transparent to programs, as it enters the pasted data as though it were typed from the keyboard.

Calling 68000 Subroutines

[[in a Mac provides the facility for a 6502 program to execute 68000 subroutines. This gives a 6502 program access to all of the Macintosh ROM routines, including Quickdraw and the Resource Manager.

CAUTION: Due to the nature of this feature, it is imperative that extreme caution be exercised when using it. It is possible to overwrite parts of the operating system or [[in a Mac itself if a 68000 subroutine goes astray!

To perform a 68000 call, the following code should be added to the program:

```
$07 $CA $AC The address of the routine  
      (The address of the routine must be even!)
```

6502 execution continues after the address value. The 68000 routine may use all of the registers except A7. Register A5 must be preserved if any Toolbox calls are to be made. The 6502 registers are passed to the subroutine in 68000 registers as follows:

- ACC is passed and returned in D1
- X is passed and returned in D2
- Y is passed and returned in D3
- S is passed and returned in D4
- P is passed and returned in D5
- A1 points to Apple memory location 0
- A2 points to the Apple stack area (A1+\$100)

This system allows a 68000 routine to perform a calculation and return the result directly to the 6502 registers.

Illegal Opcodes

When the 6502 encountered an illegal opcode, it behaved in unpredictable ways. Normally the only reason for a program to execute an undefined opcode was if it had 'crashed' or run off into a section of memory where it should not have been. If this happened, the user had no way of knowing it. With][in a Mac, this problem has been solved.

Anytime that an undefined opcode is encountered, the following box is displayed:



**The program that you are
running is attempting to execute
an undefined 6502 instruction
at address \$0300
It has probably 'crashed'.**

Resume

Reset

Monitor

Re-start

At this point the user has four options. 'Resume' goes on with the next instruction as though nothing had happened. 'Reset' is equivalent to pressing RESET (command-period). 'Monitor' causes the Apple monitor to become active. 'Re-start' is used if a program has totally "locked up" the machine; it re-loads all of the internal code and re-boots the current disk.

If this box appears while writing software, it generally means that the program has branched out of its intended area. The address displayed in the box above can many times help to determine the cause.

Appendix A: References

Recommended Reading

- Apple Computer, Inc. *Apple II Reference Manual* . Jan. 1979.
- Apple Computer, Inc. *Apple II Basic Programming Manual* . 1978.
- Apple Computer, Inc. *Applesoft Tutorial*. 1983.
- Apple Computer, Inc. *Applesoft Basic Programmer's Reference Manual*. 1982,1983.
- Apple Computer, Inc. *Basic Programming with ProDOS*. 1984.
- Apple Computer, Inc. *ProDOS Technical Reference Manual*. 1985.
- Apple Computer, Inc. *The DOS Manual*. 1980.
- Quality Software. *Beneath Apple DOS*. Don Worth and Pieter Lechner. 1981.
- Quality Software. *Beneath Apple ProDOS*. Don Worth and Pieter Lechner. 1984.
-

Trademarks

The following is a list of Registered Trademarks referred to in this manual.

Apple	Apple Computer, Inc.
Apple II	Apple Computer, Inc.
Apple II+	Apple Computer, Inc.
Applesoft	Apple Computer, Inc.
Integer Basic	Apple Computer, Inc.
Macintosh	Apple Computer, Inc.
ProDOS	Apple Computer, Inc.
DOS 3.3	Apple Computer, Inc.
[[in a Mac	COMPUTER:applications, Inc.

Appendix B: Applesoft Reference

This chapter contains a list of all of the applesoft commands, with a brief description of each one. More complete information can be found in the *Applesoft Basic Programmer's Reference Manual* or the *Applesoft Tutorial*, both by Apple Computer.

Statements and Lines

A line typed without a line number is executed immediately; those lines typed with a line number are saved for execution with the RUN command.

:	Separates multiple statements on the same line.
REM	Designates a remark for comments.

Operations on Whole Programs

NEW	Erases the current program and clears all variables.
CLEAR	Resets all variables.
LIST	Displays the entire current program.
LIST 11-12	Displays the lines from 11 to 12.
RUN	Executes the current program.
RUN n	Executes the current program from line n.
RUN <i>name</i>	Executes the program <i>name</i> from disk.
LOAD <i>name</i>	Loads the program <i>name</i> from disk.
SAVE <i>name</i>	Saves the program on the disk as <i>name</i> .

Variables

Real	AB	+/- 9.9999999 E+37
Integer	AB%	+/- 32767
String	AB\$	0 to 255 characters

Where A is a letter, B is a letter or digit.

Arrays

Real	AB(x,y,z)
Integer	AB%(x,y,z)
String	AB\$(x,y,z)

Where A is a letter, B is a letter or digit. The size of an array is limited only by the available memory.

DIM a (x,y,z) Defines the array a with maximum subscripts of x, y and z.

Arithmetic Operators

=	Assigns a value to a variable
-	Subtraction
+	Addition
/	Division
*	Multiplication
^	Exponentiation

Logical Operators

AND	Both true
OR	Either one or both true
NOT	Is false

Relational Operators

=	Equal to
<	Less than
>	Greater than
<= =<	Less than or equal to
>= =>	Greater than or equal to
<> ><	Not equal to

Arithmetic Functions

ABS(x)	Absolute value of x
SGN(x)	-1 if x<0, 0 if x=0, 1 if x>0
INT(x)	Integer portion of x
SQR(x)	Square root of x
SIN(x)	Sine of x
COS(x)	Cosine of x
TAN(x)	Tangent of x
ATN(x)	Arctangent of x
EXP(x)	e raised to the xth power
LOG(x)	Natural logarithm of x
RND(x)	if x>0, random number between 0 and 1 if x=0, repeats last random number if x<0, begins new repeatable sequence
DEF FN (x) = expr	Defines a function

String Operations

+	Concatenates Strings
LEN(s)	Length of string s
LEFT\$(s,x)	Leftmost x characters of string s
MID\$(s,x,y)	y characters from s, beginning at position x
RIGHT\$(s,x)	Rightmost x characters of string s
STR\$(x)	String representing x
VAL(s)	Numeric value of string s
CHR\$(x)	Character with ASCII code x
ASC(s)	ASCII code for first character in string s

Control

GOTO n	Branches to line n
ON expr GOTO n1,n2,n3...	Branches to line n1,n2,n3... depending on the value of expr
IF cond THEN s1:s2:s3...	Executes statements s1,s2,s3... if expr is true
FOR v = x TO y STEP z	Begins a loop for all values of v from x to y by z; if step is omitted, 1 is understood.
NEXT v	Repeats loop for next value of v
GOSUB n	Branches to subroutine at line n
RETURN	Returns to point of call from a subroutine
ON expr GOSUB n1,n2,n3...	Branches to subroutine at line n1,n2,n3... depending on the value of expr
POP	Removes one address from the return stack
ONERR GOTO n	Sets the line number branched to when an error occurs.
RESUME	Re-executes statement causing an error
STOP	Halts program and print line number
CONT	Resumes program execution
END	Halts program execution

Utility Statements

PEEK(addr)	Value of memory location addr
POKE addr,x	Sets memory location addr to x
CALL addr	Executes machine language routine at addr
USR(x)	Passes argument to machine language routine
HIMEM:addr	Sets highest available memory to addr
LOMEM:addr	Sets lowest available memory to addr
FRE(0)	Amount of available storage
TRACE	Displays the number of each executed line
NOTRACE	Turns line numbering off

Graphics

GR	Sets lores graphics mode and clears screen
COLOR=x	Sets lores drawing color to x
PLOT x,y	Draws a dot at location x,y
HLIN x1,x2 at y	Draws a horizontal line from x1,y to x2,y
VLIN y1,y2 at x	Draws a vertical line from x,y1 to x,y2
SCRN(x,y)	Color on the screen at x,y
HGR	Displays hires page1, mixed mode.
HGR2	Displays hires page2, full screen graphics.
HCOLOR=x	Sets hires drawing color to x
HPLOT x,y	Plots a dot at coordinate x,y
HPLOT x1,y1 TO x2,y2...	Draws a hires line from x1,y1 to x2,y2...
DRAW n AT x,y	Draws shape n at coordinate x,y
XDRAW n AT x,y	Draws shape n at x,y using exclusive-or
SCALE=x	Sets scale to x for shape drawing
ROT=x	Sets rotation to x for shape drawing

Input/Output

IN# n	Re-directs input from slot number n
INPUT s;x,y,...	Prompts with string s, then reads x,y,...
GET c	Reads a single character from keyboard
READ x,y,...	Reads values from DATA list into x,y,...
DATA x,y,...	List of data value
RESTORE	Restarts DATA list from beginning
PDL(n)	Value of paddle n, either 0 or 1
PR# n	Re-directs output to slot number n
PRINT x,y,...	Prints values x,y,...
TEXT	Sets text mode
HOME	Clears the text screen
HTAB x	Sets cursor horizontal to x
VTAB x	Sets cursor vertical to x
INVERSE	Sets text printing to black on white
NORMAL	Sets text printing to white on black

Appendix C: Integer Basic Reference

This chapter contains a list of the Integer Basic commands, and a brief description of the function of each. For more detailed descriptions see *APPLE II BASIC Programming Manual*.

Statements and Lines

A line typed without a line number is executed immediately; those lines typed with a line number are saved for execution with the RUN command.

:	Separates multiple statements on the same line.
REM	Designates a remark for comments.

Operations on Whole Programs

NEW	Erases the current program and clears all variables.
CLEAR	Resets all variables.
LIST	Displays the entire current program.
LIST 11-12	Displays the lines from 11 to 12.
RUN	Executes the current program.
RUN n	Executes the current program from line n.
RUN <i>name</i>	Executes the program <i>name</i> from disk.
LOAD <i>name</i>	Loads the program <i>name</i> from disk.
SAVE <i>name</i>	Saves the program on the disk as <i>name</i> .

Variables

Integer	AB	+/- 32767
String	AB\$	0 to 255 characters

Where A is a letter, B is a letter or digit

Arrays

DIM	Allocates space for an array or string.
-----	---

Arithmetic Operators

=	Assigns a value to a variable.
-	Subtraction.
+	Addition.
/	Division.
*	Multiplication.
^	Exponentiation.

Logical Operators

AND	Both true
OR	Either one or both true
NOT	Is false

Relational Operators

=	Equal to
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
#	Not equal to

Arithmetic Functions

ABS(x)	Absolute value of x
SGN(x)	-1 if x<0, 0 if x=0, 1 if x>0
RDN(x)	Random number in the range $0 \leq r < x$

String Operations

ASC(s)	ASCII code for first character in string s.
LEN(s)	Length of string s.

Control

GOTO n	Branches to line n.
IF cond THEN s1	Executes statement if expr is true.
FOR v = x TO y STEP z	Begins a loop for all values of v from x to y by z; if step is omitted, 1 is understood.
NEXT v	Repeats loop for next value of v.
GOSUB n	Branches to subroutine at line n.
RETURN	Returns to point of call from subroutine.
POP	Removes one address from the return stack.
STOP	Halts program and prints line number.
CON	Resumes program execution.
END	Halts program execution.

Utility Functions

Peek(addr)	Value of memory location at addr.
POKE addr,x	Sets memory location addr to x.
CALL addr	Executes a machine language routine at addr.
HIMEM:addr	Sets highest available memory to addr.
LOMEM:addr	Sets lowest available memory to addr.
TRACE	Displays the number of each executed line.
NOTRACE	Disables the TRACE mode.
DSP v1,v2..	Displays v1,v2... each time they change.

Graphics

GR	Sets lores graphics mode and clears screen.
COLOR=x	Sets lores color to x.
PLOT x,y	Draws a dot at location x,y.
HLIN x1,x2 at y	Draws a horizontal line from x1,y to x2,y.
VLIN y1,y2 at x	Draws a vertical line from x,y1 to x,y2.
SCRN(x,y)	The color of the dot at location x,y.

Input/Output

IN# n	Re-directs input from slot n.
INPUT s,x,y	Prompts with string s, then reads x,y.
PDL(n)	Value of paddle n, either 0 or 1.
PR# n	Re-directs output to slot n.
PRINT x,y...	Prints strings or values.
TEXT	Sets text mode.
VTAB n	Sets vertical position to n.

Appendix D: DOS 3.3 Reference

This appendix provides a brief description of the commands available with the DOS 3.3 operating system. For more detailed information see *The DOS Manual*.

BASIC Program Commands

FP	Activates applesoft basic and clears any program currently in memory.
INT	Activates integer basic and clears any program currently in memory.
RUN name	Executes 'name' from disk.
LOAD name	Loads 'name' from disk.
SAVE name	Saves the current program as 'name' on disk.
INIT name	Initializes a disk, erasing all data from it and setting the startup program to 'name'.
MAXFILES n	Sets the maximum number of open files.
MON {,C} {,I} {,O}	Selects whether Commands, Output, or Input are echoed to the screen.
NOMON {,C} {,I} {,O}	Disables printing of DOS operations.

Programming Commands

CHAIN name	Allows integer basic programs to run other. integer basic programs.
PR# n	Re-directs output to slot n.
IN# n	Re-directs input from slot n.

Filing Commands

CATALOG	Displays the current disk directory.
RENAME	oldname,newname Changes the name of a disk file.
DELETE	name Deletes the file 'name'.
LOCK	name Prevents the file 'name' from being delted, renamed or changed.
UNLOCK	name Removes file locking.
VERIFY	name Checks to see that an entire file is readable.

Text File Commands

OPEN	name Opens a text file for reading.
CLOSE	name Closes a text file.
READ	name Sets the file 'name' to read mode.
WRITE	name Sets the file 'name' to write mode.
APPEND	name Sets write mode for 'name' at the end of the curent file.
POSITION	name,x Moves the record pointer for 'name' to the position specified by x.
EXEC	name Reads text from file 'name' entering it as though it were typed from the keyboard.

Binary File Commands

BRUN	name Executes the binary file 'name'.
BSAVE	name ,A ,L Saves the file 'name' starting at A, with length L.
BLOAD	name Loads the binary file 'name'.

Appendix E: ProDOS Reference

This appendix provides a brief description of the various commands of the ProDOS operating system. For more detailed information see *BASIC Programming with ProDOS* or *ProDOS Technical Reference Manual*.

Binary File Commands

BRUN name Executes the binary file 'name' from disk.

BSAVE name,A,L

Saves the binary file 'name', starting at address A, length L to disk.

BLOAD name Loads the binary file 'name' from disk.

BASIC Program Commands

- (dash) Executes a BASIC, binary, or EXEC file.

RUN name Executes the BASIC program name.

LOAD name Loads the file name from disk.

SAVE name Saves the current program as name.

Programming Commands

CHAIN name Allows one program to run another.

STORE name Stores all basic variables to disk.

RESTORE name

Restores all basic variables from disk.

PR# n Re-directs character output to slot n.

IN# n Re-directs character input from slot n.

FRE Performs applesoft house cleaning.

BYE Exits current program.

Text File Commands

OPEN name	Opens a text file on disk.
CLOSE name	Close a text file on disk.
READ name	Sets read mode for name.
WRITE name	Sets write mode for name.
APPEND name	Sets write mode at the end of name.
FLUSH name	Writes any unwritten information to name.
POSITION name	Moves the file pointer of name.
EXEC name	Reads text from a file, passing it to the . current program as though it were typed from the keyboard.

Filing Commands

CAT	Displays the current disk directory (40 col).
CATALOG	Displays the current disk directory (80 col).
PREFIX pathname	Displays the current prefix, or sets the prefix.
CREATE name	Creates a new disk file.
RENAME oldname,newname	Changes the name of a disk file.
DELETE name	Removes a file from the disk drive.
LOCK name	Prevents a file from being deleted, renamed, or changed.
UNLOCK name	Removes file locking.
VERFIY name	Verifies that a particular file exists.

Cable Connections

Shown below are the proper connections for wiring a cable between an Apple II and II in a Mac. Most imagewriter cables will work with the Super Serial Card for performing transfers, although some early cables are missing some pins. The cables shown below may work for other serial cards if they are similar to those mentioned below.

